

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Previously presented) A method for moving a data object in a computer system from a first to a second storage location, the method comprising:
 - a) selecting a data object from the first storage location;
 - b) assigning at least one identifier (ID) to the selected data object;
 - c) storing the at least one ID in at least one lock object;
 - d) storing the selected data object at the second storage location and associating the second storage location with the at least one ID in the at least one lock object;
 - e) deleting the selected data object from the first storage location;
 - f) deleting the at least one ID from the at least one lock object after the storing of the data object in the second storage location has been completed; and
 - g) checking before or while performing any of steps a) to c), whether the at least one ID has been stored in the at least one lock object and, if yes, skipping at least step c).
2. (Original) The method of claim 1, wherein
the data object comprises one or more fields of one or more tables and wherein the at least one ID comprises one or more key fields of the one or more tables.
3. (Original) The method of claim 1, comprising:
storing the second storage location in the at least one lock object.

4. (Original) The method of claim 2, comprising:
storing the second storage location in the at least one lock object.

5. (Previously presented)

A method for moving a data object in a computer system from a first to a second storage location, the method comprising:

- a) selecting a data object from the first storage location;
- b) assigning a first type of ID and a second type of ID to the data object;
- c) storing the first type of ID in a first lock object;
- d) storing the second type of ID in a second lock object;
- e) storing the data object at the second storage location and associating the second storage location with the at least the first type of ID;
- f) deleting the first type of ID from the first lock object after the data object has been deleted from the first storage location; and
- g) deleting the second type of ID from the second lock object after the data object has been stored in the second storage location.

6. (Original) The method of claim 5, wherein
the second type of ID is stored in the second lock object immediately after assigning IDs to the data object.

7. (Original) The method of claim 5, wherein
the second type of ID is stored in the second lock object before the storing of the data object at the second storage location is started.

8. (Original) The method of claim 5, wherein
the first type of ID is stored in the first lock object before the storing of the data object at
the second storage location is started.

9. (Original) The method of claim 6, wherein
the first type of ID is stored in the first lock object before the storing of the data object at
the second storage location is started.

10. (Original) The method of claim 7, wherein
the first type of ID is stored in the first lock object before the storing of the data object at
the second storage location is started.

~~11.~~ (Canceled).

¹³
~~12.~~ (Original) The method of claim 5, further comprising:
checking before or while performing any of steps a) to d), whether the data object is
stored in the second storage location and, if yes, skipping at least step d).

¹¹
~~13.~~ (Currently amended) The method of claim 1 44, wherein
the checking is performed by querying a lock object.

¹³
14. (Original) The method of claim ~~12~~, wherein
the checking is performed by querying a lock object.

¹²
~~15.~~ (Currently amended) The method of claim 1 44, further comprising:
in case of a failure in step d), checking whether the data has been completely stored in

the second storage location and, if the data has not been completely stored, skipping at least steps e) and f).

16. (Previously presented) A computer system for moving a data object from a first to a second storage location, the system comprising:

memory having program instructions;

storage means for storing data;

at least one processor to execute the program instructions to perform operations comprising:

- a) selecting a data object from the first storage location;
- b) assigning at least one identifier (ID) to the selected data object;
- c) storing the at least one ID in at least one lock object;
- d) storing the data object at the second storage location and associating the second storage location with the at least one ID in the at least one lock object;
- e) deleting the data object from the first storage location;
- f) deleting the at least one ID from the at least one lock object after the storing of the data object in the second storage location has been completed; and
- g) checking before or while performing any of steps a) to c), whether the at least one ID has been stored in the at least one lock object and, if yes, skipping at least step c).

17. (Original) The computer system of claim 16, wherein the data object comprises one or more fields of one or more tables and wherein the at least one ID comprises one or more key fields of the one or more tables.

18. (Original) The computer system of claim 16, wherein the operations performed by the processor comprise:
storing the second storage location in the at least one lock object.

19. (Previously presented) The computer system of claim 17, wherein the operations performed by the processor comprise storing the second storage location in the at least one lock object.

20. (Previously presented) A computer system for moving a data object from a first to a second storage location, the system comprising:
memory storing program instructions;
storage means for storing data;
at least one processor to execute the program instructions to perform operations comprising:

- a) selecting a data object from the first storage location;
- b) assigning a first type of ID and a second type of ID to the data object;
- c) storing the first type of ID in a first lock object;
- d) storing the second type of ID in a second lock object;
- e) storing the data object at the second storage location and associating the second storage location with the at least the first type of ID;
- f) deleting the first type of ID from the first lock object after the data object has been deleted from the first storage location; and

g) deleting the second type of ID from the second lock object after the data object has been stored in the second storage location.

21. (Original) The computer system of claim 20, wherein the second type of ID is stored in the second lock object immediately after assigning IDs to the data object.

22. (Original) The computer system of claim 20, wherein the second type of ID is stored in the second lock object before the storing of the data object at the second storage location is started.

23. (Original) The computer system of claim 20, wherein the first type of ID is stored in the first lock object before the storing of the data object at the second storage location is started.

24. (Original) The computer system of claim 21, wherein the first type of ID is stored in the first lock object before the storing of the data object at the second storage location is started.

25. (Original) The computer system of claim 22, wherein the first type of ID is stored in the first lock object before the storing of the data object at the second storage location is started.

~~26. (Canceled).~~

28

~~27~~. (Original) The computer system of claim 20, further comprising:

checking before or while performing any of steps a) to d), whether the data object is stored in the second storage location and, if yes, skipping at least step d).

26

~~28~~. (Currently amended) The computer system of claim 16 26, wherein the checking is performed by querying a lock object.

28

29. (Original) The computer system of claim ~~27~~, wherein the checking is performed by querying a lock object.

27

~~30~~. (Currently amended) The computer system of claims 16 26, further comprising:

in case of a failure in step d), checking whether the data has been completely stored in the second storage location and, if the data has not been completely stored, skipping at least steps e) and f).

storage

31. (Previously presented) A computer-readable ^{storage} medium comprising instructions that, when executed by a processor, move data objects in a computer system from a first to a second storage location, the medium comprising instructions for:

- a) selecting a data object from the first storage location;
- b) assigning at least one identifier (ID) to the selected data object;
- c) storing the at least one ID in at least one lock object;
- d) storing the data object at the second storage location and associating the second storage location with the at least one ID in the at least one lock object;
- e) deleting the data object from the first storage location;

f) deleting the at least one ID from the at least one lock object after the storing of the data object in the second storage location has been completed; and

g) checking before or while performing any of steps a) to c), whether the at least one ID has been stored in the at least one lock object and, if yes, skipping at least step c).

32. (Previously presented) A computer-readable ^{storage} medium storing instructions that, when executed by a processor, move at least one data object in a computer system from a first to a second storage location, the medium storing instructions for:

- a) selecting at least one data object from the first storage location;
- b) assigning a first type of ID and a second type of ID to the at least one data object;
- c) storing the first type of ID in a first lock object;
- d) storing the second type of ID in a second lock object;
- e) storing the at least one data object at the second storage location and associating the second storage location with at least the first type of ID;
- f) deleting the first type of ID from the first lock object after the at least one data object has been deleted from the first storage location; and
- g) deleting the second type of ID from the second lock object after the at least one data object has been stored in the second storage location.

~~33. (Canceled).~~

15
~~24~~. (Previously presented) A method for moving a data object in a computer system from a first to a second storage location, the method comprising:

- a) selecting a data object having at least one identifier (ID) from the first storage location;
- b) storing the at least one ID in at least one lock object;
- c) storing the data object at the second storage location and associating the second storage location with the at least one ID in the at least one lock object;
- d) deleting the data object from the first storage location; and
- e) deleting the at least one ID from the at least one lock object after the data object has been deleted from the first storage location;
- f) deleting the at least one ID from the at least one lock object after the storing of the data object in the second storage location has been completed; and
- g) checking before or while performing steps a) or b), whether the at least one ID has been stored in the at least one lock object and, if yes, skipping at least step b).

30
~~35~~. (Original) A method for a moving data object in a computer system from a first to a second storage location, the method comprising:

- a) selecting a data object from the first storage location;
- b) assigning an identifier (ID) of a first type to the selected data object;
- c) assigning an ID of a second type to the selected data object;
- d) storing the first type ID in a first lock object;
- e) storing the second type ID in a second lock object;
- f) storing the data object at the second storage location;

- g) deleting the data object from the first storage location;
- h) deleting the first type ID from the first lock object after the storing of the data object in the second storage location has been completed; and
- i) deleting the second type ID from the second lock object after the storing of the data object in the second storage location has been completed.

33

- ~~36~~. (Previously presented) A method for moving a data object in a computer system from a first to a second storage location, the method comprising:
- a) selecting a data object having at least one identifier (ID) from the first storage location;
 - b) storing the at least one ID in a first lock object;
 - c) storing the at least one ID in a second lock object;
 - d) storing the data object at the second storage location;
 - e) deleting the data object from the first storage location; and
 - f) deleting the at least one ID from the first lock object after the data object has been stored at the second storage location; and
 - g) checking before or while performing steps a) or b), whether the at least one ID has been stored in the at least one lock object and, if yes, skipping at least step b).